

IN THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1-7. (canceled)

8. (Previously presented) A computer program for specification compatibility tracking for a software product, comprising:

a code segment that associates testable assertions with statements within an input specification;

5 a code segment that associates a context with each of the testable assertions, wherein the input specification has a multi-level tree structure having a plurality of nodes, each of the statements being contained by one of the nodes, wherein the context for each of the testable assertions is defined as the node containing the statement associated with the testable assertion;

10 a code segment that binds each of the testable assertions to one of a plurality of tests that test the testable assertion, each one of the tests being a computer program testing an implementation of the software product to determine whether the software product complies with the portion of the input specification that corresponds with the testable assertion bound to the one of the tests;

15 a code segment that identifies each testable assertion as tested, non-tested, or invalid; and

a code segment for presenting information on coverage of the input specification by tests.

9-20 (Canceled)

21. (Previously presented) The computer program of claim 8, wherein the information includes a percentage of the testable assertions covered by the tests.

22. (Previously presented) The computer program of claim 8, wherein the information includes a list of the testable assertions tested by the tests.

23. (Previously presented) The computer program of claim 8, wherein the information includes a list of the testable assertions not tested by the tests.

24. (Previously presented) The computer program of claim 8, further comprising:
a code segment identifying ones of the testable assertions affected by a change to the input specification.

25. (Previously presented) A method for specification compatibility tracking for a software product, comprising:

associating testable assertions with statements within an input specification;

5 associating a context with each of the testable assertions, wherein the input specification has a multi-level tree structure having a plurality of nodes, each of the statements being contained by one of the nodes, wherein the context for each of the testable assertions is defined as the node containing the statement associated with the testable assertion;

10 binding each testable assertion to one of a plurality of tests that test the testable assertion, each one of the tests being a computer program testing an implementation of the software product to determine whether the software product complies with the portion of the input specification that corresponds with the testable assertion bound to the one of the tests;

identifying each testable assertion as tested, non-tested, or invalid; and

presenting information on coverage of the input specification by tests.

26. (Previously presented) The method of claim 25, wherein the information includes a percentage of the testable assertions covered by the tests.

27. (Previously presented) The method of claim 25, wherein the information includes a list of the testable assertions tested by the tests.

28. (Previously presented) The method of claim 25, wherein the information includes a list of the testable assertions not tested by the tests.

29. (Previously presented) The method of claim 25, further comprising:

identifying ones of the testable assertions affected by a change to the input specification.

30. (Currently amended) The computer program of claim 8 wherein the tree structure comprises a top-level specification, package-level specifications as immediate children of the top level specification, class-level specifications as children of the package-level specifications, and leaves comprising one or more of a constructor specification, a method specification, and a field specification ~~constructor/method/field specifications as leaves~~, each of the top-level specification, package-level specification, class-level specification and ~~constructor/method/field specifications~~ leaves being identifiable as one of the nodes in the tree-like structure.

31. (Currently amended) The method of claim 25 wherein the tree structure comprises a top-level specification, package-level specifications as immediate children of the top level specification, class-level specifications as children of the package-level specifications, and leaves comprising one or more of a constructor specification, a method specification, and a field specification ~~constructor/method/field specifications as leaves~~, each of the top-level specification, package-level specification, class-level specification and ~~constructor/method/field specifications~~ leaves being identifiable as one of the nodes in the tree-like structure.